THE SAN DIEGO ATARI® COMPUTER ENTHUSIASTS NEWSLETTER

JUNE 1982

VTOC

Sounds of Atari.....2
Code Conversion.....3,4,5
North County Club.....5
Multiple Screens.....5
Hardware Interface.....6,7
FORTH Class.....7

San Diego ATARI
Computer Enthusiasts

is an independent non-profit computer organization and users' group, with no connection to ATARI Corp., a Warner Communications Co. Membership fees are currently $5.00 annually from January 1982 to January 1983. Membership includes free access to the SD-ACE computer program library, subscription to the SD-ACE NEWSLETTER, and classes when held. Permission to reprint articles in any non-commercial publication is permitted, without written authorization, as long as proper credit is given.

SD-ACE OFFICERS

President	Dick Hiatt	463-8460
Vice Pres	Walt Snell	569-6584
Treasurer	James Marr	272-6010
Secretary	Don Perkins	479-6253
Librarian	Chuck Fowler	447-8143
Editor	Ron Miller	748-7195
At-large	Dave Story	450-9043

Correspondence Address: C/O Dick Hiatt - President, 5353 Baltimore Drive #39, La Mesa, Ca. 92041

Newsletter Address: C/O Ron Miller, 12945 Grimsley Ave, Poway, Ca. 92064

Supporting Librarians:
Jim Gollwitzer (Poway) 486-2479
Nestor Sanchez (PBeach) 488-3325

FROM THE PREZ...

Our club is realy growing. We now have about 158 members as of last Tuesday. Ron Miller is starting a North County Sub Group which will be supported by your club. We all will have a common newsletter and program library. We are not as big as M.A.C.E. (500 plus members) but size is not our goal. What we are lacking is more support from our members in submitting newsletter articles and original programs to our library. We cannot trade with other clubs unless we have something original to give them. Let us decide if it is good or bad, just send stuff in.For those of you that did not get to the last meeting, we passed around a membership list for you to indicate if you wanted to have your name placed on mailing list of computer product suppliers. If you did not get a chance to indicate that you would please call me and let me know by the next meeting. We will only send out the members names that say it is OK. The Board of Directors decided that we would have a booth at the next 1983 SAN DIEGO COMPUTER EXPO which will be Feb. 18-21 1983. We will be asking for volunteers to be on a booth committee. If you are interested please see me. I am also looking for a program chairman and ideas for programs for the meetings. This month we are going to have general discussion and demonstration of programs. If you have any questions or problems bring them to the meeting and we will try to find some help. Also bring any new software (good or bad) you would like to demo. Software or hardware demos help all of us. WILL THE PERSON WHO VOLUNTEERED TO HELP WITH THE BUSNESS CARDS PLEASE CALL ME. I DID NOT GET YOUR NAME IN THE CONFUSION OF THE LAST MEETING. See you all at the next meeting.

FROM THE LIBRARY...

"In the spring a young mans fancy turns to programing the program of programs." Seriously, I'm amazed at the growth of our little (?) club in the past two years. I would like to welcome all our new members, who made it possible. I would also like to encourage our new members to make use of the services the club provides. Along these lines, in the past year I have copied a lot of programs for you folks out there and, while looking for something else, I came across some media that had been left for copying. The items never got copied for one reason or another. So Mr. Fedchenco (hope I spelled that right) on the disk you left me, what do you want on it? And to a husband and wife (wifes name Yvonne), who left me a tape, what do you want on it? If your memory is better than mine give the main library a call and inform me. For the people who were at the last meeting and saw my robot but did not see it demonstrated, I will be bringing it to the June meeting. I will also be demonstrating a series of educational programs from Santa Cruz Software. These programs greatly simplify page flipping, animation, and p/m graphics. I would like to apoligize to the people who were called by Luth Reasearch, and did not want to be contacted. I provided them with some names in the hope that we could make our gripes known to Atari, who I think authorized the survey.

NEW PROGRAMS...

SOLITAIR,D15,24k,BASIC
Computer version of the old card game, very good.

SEARCH,D16,16k,basic, Word search puzzle

CROSSREF,D26,24k,basic,DISK ONLY
Find where and how many variables you have in your program.

Thanks to Jim Golwiser for the following:

WOW,D4,8k,basic, GTIA demo.
WOWII,D4,8k,basic, GTIA demo
PGHOST2,D4,8k,basic, P/M demo
MONITOR,D26,4k,basic, Follow memory from whatever place you start.
MPG,D26,4k,basic, Figure your gas milege.
BIRTHDAY,D20,4k,basic, Birthday song.
CONCENTRATION,D10,8k,basic, the popular TV and board game.

********** FOR SALE **********
ATARI 825 PRINTER
80 COLUMN
CABLES & MANUALS.....$400.00
Call Chuck at 447-7072
****

SOUNDS OF ATARI

Since I did not get any graphics input from the club there will be no graphics corner, however, there will be a new corner called SOUNDS. This is something everyone can enjoy since every Atari can make the same sounds and sound programs tend to be short. Sounds are fun experiments and sometimes the most unusual sounds are accidents. I will leave it to the books and magazines to explain the sound channels in the Atari. What we need to do is publish little programs that produce sounds. Even the kids can write some sound programs. Summer is here and the kids have more time to experiment, so put them to work. To start making sounds, here is a program written by Richard Kruse and published in Creative Computing. It sounds like a whistle and a bomb blast. Try it.

```
1 REM WHISTLE AND BOMB SOUNDS
2 REM DUR IS DURATION OF BOMB
5 DUR=10
10 GOSUB 100
20 END
100 REM BOMB
110 V1=4:FOR NT=30 TO 75
120 SOUND 0,NT,10,V1
130 SOUND 1,NT+3,10,V1*0.7
140 FOR WT=1 TO DUR*3:NEXT WT
150 V1=V1*1.03:NEXT NT
160 SOUND 2,35,8,12
170 V1=15:V2=15:V3=15
180 NT=DUR+5:ICR=0.79+DUR/100
190 SOUND 0,NT,8,V1
200 SOUND 1,NT+20,8,V2
210 SOUND 2,NT+50,8,V3
220 V1=V1*ICR
230 V2=V2*(ICR+0.05)
240 V3=V3*(ICR+0.08)
250 IF V3>1 THEN 190
260 SOUND 0,0,0,0:SOUND 1,0,0,0
270 SOUND 2,0,0,0:RETURN
```

CODE CONVERSION

The first thing I did after getting my Atari was to make a code conversion chart which related hex, binary, decimal, Atari's keyboard and 6502 assembly language codes. After two years of use, this chart is still the most referenced Atari information I use. You need it for printer control codes, assembly language programming, memory peeking and poking, and setting sound registers. As you gain experience with your Atari you will discover just how handy this chart is. So on the next 2.3 pages this chart is reproduced for your use. Have fun.

Ron Miller

DEC	HEX	BINARY	KEY	6502 CODE
0	0	00000000	C ,	BRK
1	1	00000001	C A	ORA (ind,X)
2	2	00000010	C B	
3	3	00000011	C C	
4	4	00000100	C D	
5	5	00000101	C E	ORA Ø page
6	6	00000110	C F	ASL Ø page
7	7	00000111	C G	
8	8	00001000	C H	PHP
9	9	00001001	C I	ORA imm
10	A	00001010	C J	ASL accum
11	B	00001011	C K	
12	C	00001100	C L	
13	D	00001101	C M	ORA abs
14	E	00001110	C N	ASL abs
15	F	00001111	C O	
16	10	00010000	C P	BPL rel
17	11	00010001	C Q	ORA (ind),Y
18	12	00010010	C R	
19	13	00010011	C S	
20	14	00010100	C T	
21	15	00010101	C U	ORA Ø p.,X
22	16	00010110	C V	ASL Ø p.,X
23	17	00010111	C W	
24	18	00011000	C X	CLC
25	19	00011001	C Y	ORA abs,Y
26	1A	00011010	C Z	
27	1B	00011011	e esc	
28	1C	00011100	eC -	
29	1D	00011101	eC =	ORA abs,X
30	1E	00011110	eC +	ASL abs,X
31	1F	00011111	eC *	
32	20	00100000	space	JSR abs
33	21	00100001	!	AND (ind,X)
34	22	00100010	"	
35	23	00100011	#	
36	24	00100100	$	BIT Ø page
37	25	00100101	%	AND Ø page
38	26	00100110	&	ROL Ø page
39	27	00100111	'	
40	28	00101000	(	PLP
41	29	00101001	)	AND imm
42	2A	00101010	*	ROL accum
43	2B	00101011	+	
44	2C	00101100	,	BIT abs
45	2D	00101101	-	AND abs
46	2E	00101110	.	ROL abs
47	2F	00101111	/	
48	30	00110000	0	BMI rel
49	31	00110001	1	AND (ind),Y
50	32	00110010	2	
51	33	00110011	3	
52	34	00110100	4	
53	35	00110101	5	AND Ø p.,X
54	36	00110110	6	ROL Ø p.,X
55	37	00110111	7	
56	38	00111000	8	SEC
57	39	00111001	9	AND abs,Y
58	3A	00111010	:	
59	3B	00111011	;	
60	3C	00111100	<	
61	3D	00111101	=	AND abs,X
62	3E	00111110	>	ROL abs,X
63	3F	00111111	?	
64	40	01000000	@	RTI
65	41	01000001	A	EOR (ind,X)
66	42	01000010	B	
67	43	01000011	C	
68	44	01000100	D	
69	45	01000101	E	EOR Ø page
70	46	01000110	F	LSR Ø page
71	47	01000111	G	
72	48	01001000	H	PHA
73	49	01001001	I	EOR imm
74	4A	01001010	J	LSR accum
75	4B	01001011	K	
76	4C	01001100	L	JMP abs
77	4D	01001101	M	EOR abs
78	4E	01001110	N	LSR abs
79	4F	01001111	O	
80	50	01010000	P	BVC rel
81	51	01010001	Q	EOR (ind),Y
82	52	01010010	R	
83	53	01010011	S	
84	54	01010100	T	
85	55	01010101	U	EOR Ø p.,X
86	56	01010110	V	LSR Ø p.,X
87	57	01010111	W	
88	58	01011000	X	CLI
89	59	01011001	Y	EOR abs,Y
90	5A	01011010	Z	
91	5B	01011011	S ,	
92	5C	01011100	S +	
93	5D	01011101	S .	EOR abs,X
94	5E	01011110	S *	LSR abs,X
95	5F	01011111	S -	
96	60	01100000	C .	RTS
97	61	01100001	a	ADC (ind,X)
98	62	01100010	b	
99	63	01100011	c	
100	64	01100100	d	
101	65	01100101	e	ADC Ø page
102	66	01100110	f	ROR Ø page
103	67	01100111	g	
104	68	01101000	h	PLA
105	69	01101001	i	ADC imm
106	6A	01101010	j	ROR accum
107	6B	01101011	k	
108	6C	01101100	l	JMP ind
109	6D	01101101	m	ADC abs
110	6E	01101110	n	ROR abs
111	6F	01101111	o	
112	70	01110000	p	BVS rel
113	71	01110001	q	ADC (ind),Y
114	72	01110010	r	
115	73	01110011	s	
116	74	01110100	t	

DEC	HEX	BINARY	KEY	6502 CODE
117	75	01110101	u	ADC 0 p.,X
118	76	01110110	v	ROR 0 p.,X
119	77	01110111	w	
120	78	01111000	x	SEI
121	79	01111001	y	ADC abs,Y
122	7A	01111010	z	
123	7B	01111011	C ;	
124	7C	01111100	S =	
125	7D	01111101	eS <	ADC abs,X
126	7E	01111110	e b.sp.	ROR abs,X
127	7F	01111111	e tab	
128	80	10000000	-C ,	
129	81	10000001	-C a	STA (ind,X)
130	82	10000010	-C b	
131	83	10000011	-C c	
132	84	10000100	-C d	STY 0 page
133	85	10000101	-C e	STA 0 page
134	86	10000110	-C f	STX 0 page
135	87	10000111	-C g	
136	88	10001000	-C h	DEY
137	89	10001001	-C i	
138	8A	10001010	-C j	TXA
139	8B	10001011	-C k	
140	8C	10001100	-C l	STY abs
141	8D	10001101	-C m	STA abs
142	8E	10001110	-C n	STX abs
143	8F	10001111	-C o	
144	90	10010000	-C p	BCC rel
145	91	10010001	-C q	STA (ind),Y
146	92	10010010	-C r	
147	93	10010011	-C s	
148	94	10010100	-C t	STY 0 p.,X
149	95	10010101	-C u	STA 0 p.,X
150	96	10010110	-C v	STX 0 p.,Y
151	97	10010111	-C w	
152	98	10011000	-C x	TYA
153	99	10011001	-C y	STA abs,Y
154	9A	10011010	-C z	TXS
155	9B	10011011	return	
156	9C	10011100	eS bsp	
157	9D	10011101	eS >	STA abs,X
158	9E	10011110	eC tab	
159	9F	10011111	eS tab	
160	A0	10100000	-space	LDY imm
161	A1	10100001	-!	LDA (ind,X)
162	A2	10100010	-"	LDX imm
163	A3	10100011	-#	
164	A4	10100100	-$	LDY 0 page
165	A5	10100101	-%	LDA 0 page
166	A6	10100110	-&	LDX 0 page
167	A7	10100111	-'	
168	A8	10101000	-(	TAY
169	A9	10101001	-)	LDA imm
170	AA	10101010	-*	TAX
171	AB	10101011	-+	
172	AC	10101100	-,	LDY abs
173	AD	10101101	--	LDA abs
174	AE	10101110	-.	LDX abs
175	AF	10101111	-/	
176	B0	10110000	-0	BCS rel

DEC	HEX	BINARY	KEY	6502 CODE
177	B1	10110001	-1	LDA (ind),Y
178	B2	10110010	-2	
179	B3	10110011	-3	
180	B4	10110100	-4	LDY 0 p.,X
181	B5	10110101	-5	LDA 0 p.,X
182	B6	10110110	-6	LDX 0 p.,X
183	B7	10110111	-7	
184	B8	10111000	-8	CLV
185	B9	10111001	-9	LDA abs,Y
186	BA	10111010	-:	TSX
187	BB	10111011	-;	
188	BC	10111100	-<	LDY abs,X
189	BD	10111101	-=	LDA abs,X
190	BE	10111110	->	LDX abs,Y
191	BF	10111111	-?	
192	C0	11000000	-@	CPY imm
193	C1	11000001	-A	CMP (ind,X)
194	C2	11000010	-B	
195	C3	11000011	-C	
196	C4	11000100	-D	CPY 0 page
197	C5	11000101	-E	CMP 0 page
198	C6	11000110	-F	DEC 0 page
199	C7	11000111	-G	
200	C8	11001000	-H	INY
201	C9	11001001	-I	CMP imm
202	CA	11001010	-J	DEX
203	CB	11001011	-K	
204	CC	11001100	-L	CPY abs
205	CD	11001101	-M	CMP abs
206	CE	11001110	-N	DEC abs
207	CF	11001111	-O	
208	D0	11010000	-P	BNE rel
209	D1	11010001	-Q	CMP (ind),Y
210	D2	11010010	-R	
211	D3	11010011	-S	
212	D4	11010100	-T	
213	D5	11010101	-U	CMP 0 p.,X
214	D6	11010110	-V	DEC 0 p.,X
215	D7	11010111	-W	
216	D8	11011000	-X	CLD
217	D9	11011001	-Y	CMP abs,Y
218	DA	11011010	-Z	
219	DB	11011011	-S ,	
220	DC	11011100	-S +	
221	DD	11011101	-S .	CMP abs,X
222	DE	11011110	-S *	DEC abs,X
223	DF	11011111	-S -	
224	E0	11100000	-C .	CPX imm
225	E1	11100001	-a	SBC (ind,X)
226	E2	11100010	-b	
227	E3	11100011	-c	
228	E4	11100100	-d	CPX 0 page
229	E5	11100101	-e	SBC 0 page
230	E6	11100110	-f	INC 0 page
231	E7	11100111	-g	
232	E8	11101000	-h	INX
233	E9	11101001	-i	SBC imm
234	EA	11101010	-j	NOP
235	EB	11101011	-k	
236	EC	11101100	-l	CPX abs

```
DEC  HEX  BINARY    KEY     6502 CODE
237   ED  11101101 -m       SBC abs
238   EE  11101110 -n       INC abs
239   EF  11101111 -o
240   F0  11110000 -p       BEQ rel
241   F1  11110001 -q       SBC (ind),Y
242   F2  11110010 -r
243   F3  11110011 -s
244   F4  11110100 -t
245   F5  11110101 -u       SBC 0 p.,X
246   F6  11110110 -v       INC 0 p.,X
247   F7  11110111 -w
248   F8  11111000 -x       SED
249   F9  11111001 -y       SBC abs,Y
250   FA  11111010 -z
251   FB  11111011 -C ;
252   FC  11111100 -S =
253   FD  11111101 eC 2     SBC abs,X
254   FE  11111110 eC bsp   INC abs,X
255   FF  11111111 eC >
```

NORTH SAN DIEGO COUNTY ATARI CLUB

There is an effort underway to form an Atari club in North County. Of the 70 members at the last meeting, about 20 indicated they were from North County. Since our monthly meetings are just getting too big, we are going to break into smaller clubs. Our intentions are to keep a consolidated library and newsletter. Our educational classes will also remain a combined effort. North County is just the first step and South Bay and East County may follow suit in the future. The people behind this are Ron Miller (748-7195) and Ted Langlet (485-8656). We intend to hold the North County meetings a day or 2 after the San Diego meetings. This keeps the newsletter, board meetings and club meetings spaced at nice intervals. We are looking now for a meeting place. If anyone has ideas, give one of us a call. If no one calls, no matter. Ted and I have lots of ideas. Anyway, this will be much better in the long run because smaller meetings are more fun and more personal. If any North County Atari folks want to help Ted and me, give one of us a call.

Ron Miller & Ted Langlet

ANIMATE

by Ed Alfaro-Atari Computer Assoc. of Orange County

Moving rapidly from one display to another gives the programmer the ability to create animation quickly and easily. This program moves the display pointer back and forth from text mode and graphics mode images that have been stored in two separate RAM locations.

First, by turning off ANTIC (line 20), we can build our separate images 30% faster than if the chip was on. As an added benefit, the screen is black, thereby eliminating having to watch the plotting sequence. The display appears completely drawn when we turn ANTIC back on.

Now we can build the first screen. Save the low and high values the computer assigns to GR.2 display location (line 50). What is line 40? MEMTOP is the actual top of existing RAM after storing our first screen (and is also the bottom of our first screen display data).

Line 60 lowers this border line, or "fence" by 4K of RAM, the amount needed for the next display, GR.23. The Atari RAM is incremented in "pages", each containing 256 bytes. Next, the second display is created with the usual PLOT and DRAWTO commands with XIO used to fill. Line 130 saves the values for low and high display memory locations, just as in line 50. Now we turn the screen on in line 140 and start displaying the two images by POKEing 560 and 651 with the stored values (lines 150,170).

Depending on your Atari's memory size, many more screens could be added in this way, lowering location 106 for each new screen. Don't forget to POKE 106 to its original value if your program continues.

```
10 GRAPHICS 18
15 NOW=PEEK(559):REM STORE ANTIC VALUE
20 POKE 559,0:REM TURN OFF SCREEN
30 POSITION 7,4:? #6;"STOP!!!"
40 MEMTOP=PEEK(106):REM LOCATION OF REAL MEMTOP
50 DLOW1=PEEK(560):DLHI1=PEEK(561):REM STORE DISPLAY
LOCATIONS FOR FIRST SCREEN
60 POKE 106,MEMTOP-16:REM MOVE TOP OF DISP. LOCATION DOWN
16 PAGES (4K)
70 REM DRAW SECOND SCREEN
80 GRAPHICS 23:POKE 559,0:REM TURN OFF ANTIC AGAIN
90 SETCOLOR 0,3,4:COLOR 3
100 PLOT 115,45:DRAWTO 115,25:DRAWTO 95,5:DRAWTO
65,5:DRAWTO 45,25:POSITION 45,45
110 POKE 765,1:XIO 18,#6,0,0,"S:"
120 PLOT 45,45:DRAWTO 65,65:DRAWTO 95,65:DRAWTO 115,45
130 DLOW2=PEEK(560):DLHI2=PEEK(561):REM MEM. LOCATION OF
SECOND SCREEN
140 POKE 559,NOW:REM TURN ANTIC ON
150 POKE 560,DLOW1:POKE 561,DLHI1:FOR WAIT=1 TO 50:NEXT
WAIT
160 POKE 559,0:POKE 559,NOW:REM "CLEANS UP" THE DISPLAY
170 POKE 560,DLOW2:POKE 561,DLHI2:FOR WAIT=1 TO 50:NEXT
WAIT
180 GOTO 150
```

INTERFACING YOUR ATARI TO THE REAL WORLD

By Marshall Dubin

Part One: Basic Input

With the addition of a few parts and some external circuits, your Atari personal computer can send and receive signals from the "real world." These signals can be the monitoring of alarms, thermostats, and a variety of digital and analog sources which in turn can control various motors, appliances, lights, etc. The purpose of this article is to discuss a method in which we can monitor up to four input signals from external sources in the real world.

For our purposes we will be using the joystick ports which are located on the front panel of the computer. Figure one shows a pin connection diagram of a single port. All four ports share this connection pattern.

Note that there are four pins labeled PIA. These pins are connected directly to the internal PERIPHERAL INTERFACE ADAPTER CHIP OF THE COMPUTER. The pins may be formatted for eigher input or output. We shall discuss them in more detail in future articles. Additional pins are the trigger pin for the joystick trigger buttons, two analog input pins, used for the paddle controllers, a +5v pin and a system ground pin. In this example we shall use the TRIGGER pins. Because there are four ports, each port may control a different device connected to the trigger pin of that port.

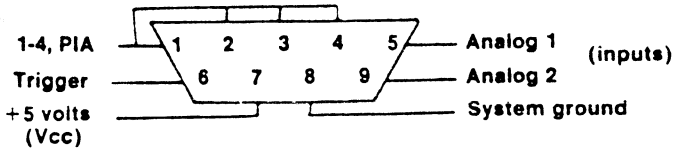

Figure 1: Pin Connections

Basic Input Using the Trigger Pin

When the computer is first switched on, the trigger pins are at a logic 1 (high) state. This is the DEFAULT status of these pins. The normal procedure when using joysticks is to have the computer react if the pin reads LOW (logic 0). When this is the case, the trigger is assumed to have been pressed. The following listing illustrates this point:

```
10 X=STRIG(0)      for port one
20 IF X=0 THEN PRINT "TRIGGER IS ACTIVE"
30 PRINT "TRIGGER IS NOT ACTIVE"
40 GOTO 20
```

The computer will take one action when the trigger is pressed and do something else when it is not. Any game using joysticks illustrates this point. Now what if an alarm sensor, liquid level sensor, or light activated switch were connected to the joystick trigger input? Ah-ha. We begin to see that the old joystick port can do more than blow away Zylons!

The Basic Input Connection

As I mentioned earlier, the default status of the trigger pin is a logic one. To simulate pressing the trigger button, we must make the status of that pin a logic zero. This is done by connecting that pin to system ground. When we do that, the pin is said to have been "pulled low," and will now read a zero.

The easiest way to do this is by connecting pin 6 of the controller port to pin 8 of the same port. In real world situations though, especially if the switch is not located right next to the computer, and especially if you are monitoring something more than a TTL level signal, this is best accomplished through the use of an external relay or better still an OPTO-ISOLATOR. Lets look at opto-isolators first.

An opto-isolator is composed of a LED (light emitting diode) and a phototransistor. When voltage is applied to the diode, it glows. The light emitted from the diode reacts with the photo sensitive transistor and "biases" it into conduction. Hefe the transistor acts as a switch which is turned on by the light from the diode, and off when the diode is dark. Since power need only be given to the diode, there is no chance that a higher damaging voltage can cross the optical barrier and damage the computer. As illustrated in figure two, when the transistor switch is ON, pin 6 of the controller port is connected to pin 8 (ground) thereby pulling it low, and simulating a trigger press.

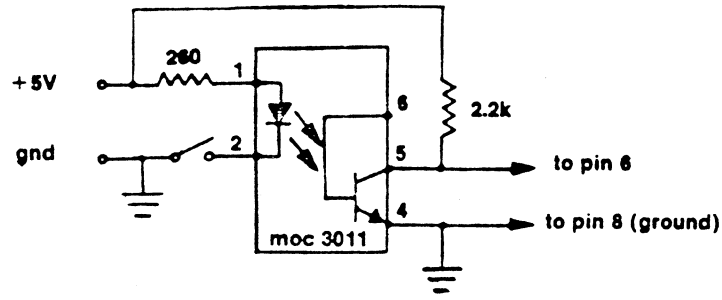

FIGURE 2: Using an opto-isolator
On this example, a relay is powered on by a high voltage such as 110 volts, and this causes the contacts to pull pin 6 low.

Do not try to power the opto-isolator from the "on board" 5 volt supply. It can only withstand 40 or so milliamps, and has other uses. Also note that the diodes in some opto-isolators can draw as much as 100 milliamps current, so you will need a supply capable of driving at lease one or more of them. In addition, you may have to use a current limiting resistor between the opto-isolator supply and the LED, especially if you use higher voltages to drive the isolator. Use Ohm's law to figure the resistance you will need for the voltage and current ratings you will be using, if they are different from mine.

Figure three shows a method of using a relay to pull the trigger pin low. Figure four shows a light activated sensor (Sargent and Shoemaker, 1981) which will also work well.

Programming Considerations

Basically, the sensors can be used in a program in pretty much the same way as the joystick trigger buttons. The keyword STRIG will read the status of this pin as it does for the normal use of the joystick. Please note that the PTRIG keyword does NOT access pin 6 of the controller port, but uses different pins. You must use the STRIG keyword, or

continued

else PEEK locations 644, 645, 646 or 647 (decimal) to read the pin status. Each location is for a different port. 644 is port one, etc. Much the same as STRIG(0).

So Now What?

Go to it! Your computer can read and react to all kinds of neat things beside Zylons or Space Invaders. Try using a light beam sensor as a counter, or determing how many times your furnace motor kicks on during the day, or reacting to a metallic "end of tape" sensor for programable slide shows, or even bio-feedback (be SURE to use optical isolation)! What it boils down to is your imagination. Experiment! Learn! Enjoy!

In future articles I will be discussing output as well as the unique built in facility of the ATARI to accept analog input without the need for complicated external circuitry. I might also suggest a very good kit made by MOSAIC ELECTRONICS FOR THE ATARI which includes two DB9S plugs, ribbon cable, and instructions for using them for interfacing. The kit is about $15.00 and is a good basic tool for the experimenters bench. Write to them at PO Box 748, Oregon City, Oregon 97045.

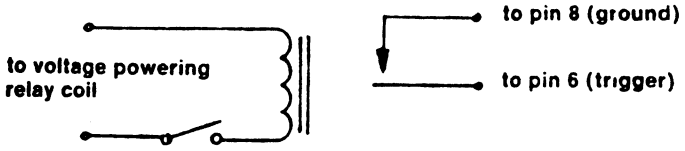

Figure 3: Use of a relay

When switch is closed, a signal is passed to the computer. Switch can be part of a relay if you wish to monitor high current devices. Have your device trigger a relay, which in turn will turn on the opto-isolator.

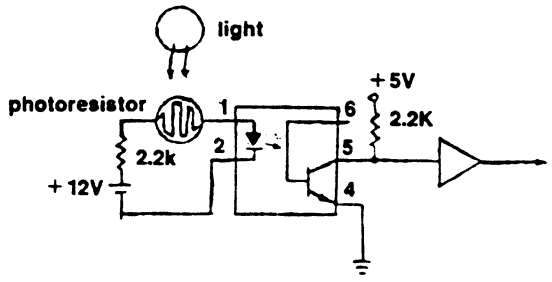

FIGURE 4

The on/off status of a light can be monitored by using a photoresistor as the LED's current limiting resistor. If the photo-resistor circuit uses more than five volts, an extra current limiting resistor may be needed (LED current shouldn't exceed 30 ma).

References

Sargent, M and Shoemaker, R, *Interfacing Microcomputers to The Real World*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1981

Mosaic Electronics Atari I/O Package, Mosaic Electronics, P.O. Box 748, Oregon City, Oregon 97045

Ⓜ

THE NEW OPERATION SYSTEM

by "Steve Sanders" - H.A.C.E.

Atari computers now coming off the assembly line have two new chips in the operating system, namely chips C#14599 and C#12499. These chips are the so-called "B" version chips. (Editor's note- they are also available from the Somerset, NJ service facility for about $20 for the pair. They go in the board just in front of your memory boards on the 800 and, therefore, are easy to install yourself.) The following are the changes in the operating system that these new chips effect:

1. Two formerly spare bytes at $0236 and $0237 are now use for the break key vector.

2. Three formerly spare bytes at $028B-$028D are used by the interrupt handler. (This means that programs that used those bytes, such as Ghost-Hunter, will not run!)

3. The IRQ interrupt handler now implements POKEY timer #4.

4. The disk drive will *no longer* temporarily stop in the middle of read-write operations. This was accomplished by changing the SETVBL routine in the NMI Interrupt handler.

5. End of buffer tests have been modified. This should eliminate false ERROR 138 messages.

P.S. It is rumored that the reason POKEY timer #4 didn't work in the original operating system was that someone made a typographical error in listing the original source code!

FORTH CLASS

I have volunteered to teach a class in FORTH if there is enough interest from our club. I define enough interest to mean 10 or more serious people who will follow through and participate in a FORTH class with me. This presents a problem. I want to know if you are interested and you don't know if you are interested because you don't know what FORTH is. So, before I take the time to make up a class I propose the following to all you potential FORTH programmers. I will organize one 2 hour class in FORTH. This class will be designed to teach you what FORTH is, what is it good for, how you can use it and where you can learn about it. This class will be good for you even if you don't intend to follow through and learn FORTH because it is very valuable to see a different approach to programming other than the BASIC, FORTRAN, COBOL, Pascal, etc, type approach. You will also learn alot about how computers are orgainzed and how languages work. Now if that interests you then let me know at the next meeting. I will have a sign up sheet. If all the above sounds like a mouth full for a short 2 hour session, you are right, it is. Get a good nights sleep before coming.

Ron Miller

Sierra Nat
5037 Ruffner
Andre Bacon

ATARI BOOKS

Dimensional Software has lots of new Atari software and they also just received a new shipment of books. They have the following books in stock.

Atari Sound & Graphics: A Self Teaching Guide, Wiley, $9.95.

Atari Games & Recreation, Reward Books, $14.95.

Atari Assembler, Inman, Reward Books, $12.95.

Your Atari Computer, Lon Poole, Osborn/Mcgraw-Hill, $15.00.

The last book is very good and almost every Atari owner is going to buy one.

Dimensional is also carrying ANTIC and Analog, which are 2 Atari magazines. And, they also are carrying GTIA chips. If you want one, they got'em.

Anyone interested in ordering any of the the following items can place orders at the regular monthly meeting.

ATARI SOFTWARE

(disk)

Apple Panic.....	$20.00	reg29.95
Fractions.....	$26.00	reg39.95
Decimals.....	$26.00	reg39.95
Compu-Read.....	$20.00	reg29.95

(cassette)

Fractions.....	$20.00	reg29.95
Decimals.....	$20.00	reg29.95
Compu-Read.....	$20.00	reg29.95
Algebra II.....	$26.00	reg39.95

Floppy Disks (box of 10)..$20.00

---

contact Dennis Costarakis at the meeting.

SAN DIEGO
ACE NEWSLETTER
5353 BALTIMORE DR #39
LA MESA, CA. 92041

COMING EVENTS

THE JUNE MEETING TAKES
PLACE ON 21 JUNE
AT 6:30 PM IN THE
SOCIAL ROOM OF THE:

NORTH PARK REC CENTER
4044 IDAHO ST
SAN DIEGO, CA

TO:

AGENDA

MUSIC DEMOS
PROGRAM DEMOS
INTEREST GROUPS
- BEGINNING ATARI
- BASIC
- ADVANCED BASIC
- FORTH
- PASCAL
- HARDWARE NUTS